Docket No. RSW920030266US1

CLAIMS:

What is claimed is:

1   1.   A method in a data processing system for collecting
2   data for analyzing memory leaks, the method comprising:
3   associating a plurality of indicators with a
4   plurality of objects, wherein the plurality of indicators
5   are set to a first state;
6   setting an indicator for each live object in the
7   plurality of objects to a second state; and
8   responsive to a request for the data, collecting
9   data from all objects in the plurality of objects having
10 indicators set to the first state.

1   2.   The method of claim 1 further comprising:
2   initiating a garbage collection process to free
3   unused objects prior to the collecting step.

1   3.   The method of claim 1, wherein each of the plurality
2   of indicators is a bit associated with a corresponding
3   object in the plurality of objects.

1   4.   The method of claim 3, wherein the first state is a
2   logic zero and the second state is a logic one.

1   5.   The method of claim 1, wherein the method is
2   implemented in a Java virtual machine.

Docket No. RSW920030266US1

1 6. The method of claim 1, wherein the data is placed
2 into a text file.

1 7. The method of claim 1, wherein the plurality of
2 objects are located in a heap.

1 8. The method of claim 1, wherein the request is
2 received from a Java application.

1 9. A method in a data processing system for collecting
2 data used to detect memory leaks in a Java virtual
3 machine, the method comprising:
4   associating an indicator with an object in a heap,
5 wherein the indicator is set to a default state;
6   responsive to a first request, setting indicators
7 for all live objects in the heap from the default state
8 to a live state; and
9   responsive to a second request, collecting
10 information for all objects having indicators in the
11 default state, wherein objects having indicators in the
12 default state are objects with memory leaks.

1 10. The method of claim 9, wherein the indicator is
2 associated with the object when the object is created.

1 11. The method of claim 9, wherein the first request and
2 the second request are received from a Java application.

Docket No. RSW920030266US1

1   12.   The method of claim 9, wherein the indicator set to

2   the live state is a logic one and the indicator set to

3   the default state is a logic zero.

1   13.   A data processing system in a data processing system

2   for collecting data for analyzing memory leaks, the data

3   processing system comprising:

4   associating means for associating a plurality of

5   indicators with a plurality of objects, wherein the

6   plurality of indicators are set to a first state;

7   setting means for setting an indicator for each live

8   object in the plurality of objects to a second state; and

9   collecting means, responsive to a request for the

10   data, for collecting data from all objects in the

11   plurality of objects having indicators set to the first

12   state.

1   14.   The data processing system of claim 13 further

2   comprising:

3   initiating means for initiating a garbage collection

4   process to free unused objects prior to initiating the

5   collecting means.

1   15.   The data processing system of claim 13, wherein each

2   of the plurality of indicators is a bit associated with a

3   corresponding object in the plurality of objects.

Docket No. RSW920030266US1

1 16. The data processing system of claim 15, wherein the
2 first state is a logic zero and the second state is a
3 logic one.

1 17. The data processing system of claim 13, wherein the
2 method is implemented in a Java virtual machine.

1 18. A data processing system in a data processing system
2 for collecting data used to detect memory leaks in a Java
3 virtual machine, the data processing system comprising:
4   associating means for associating an indicator with
5 an object in a heap, wherein the indicator is set to a
6 default state;
7   setting means, responsive to a first request, for
8 setting indicators for all live objects in the heap from
9 the default state to a live state; and
10   collecting means, responsive to a second request,
11 for collecting information for all objects having
12 indicators in the default state, wherein objects having
13 indicators in the default state are objects with memory
14 leaks.

1 19. A computer program product in a computer readable
2 medium for collecting data for analyzing memory leaks,
3 the computer program product comprising:
4   first instructions for associating a plurality of
5 indicators with a plurality of objects, wherein the
6 plurality of indicators are set to a first state;

Docket No. RSW920030266US1

7   second instructions for setting an indicator for

8 each live object in the plurality of objects to a second

9 state; and

10   third instructions, responsive to a request for the

11 data, for collecting data from all objects in the

12 plurality of objects having indicators set to the first

13 state.

1 20. The computer program product of claim 19 further

2 comprising:

3   fourth instructions for initiating a garbage

4 collection process to free unused objects prior to

5 executing the third instructions.

1 21. The computer program product of claim 19, wherein

2 each of the plurality of indicators is a bit associated

3 with a corresponding object in the plurality of objects.

1 22. The computer program product of claim 21, wherein

2 the first state is a logic zero and the second state is a

3 logic one.

1 23. The computer program product of claim 19, wherein

2 the method is implemented in a Java virtual machine.

1 24. A computer program product in a computer readable

2 medium in a data processing system for collecting data

3 used to detect memory leaks in a Java virtual machine,

4 the computer program product comprising:

Docket No. RSW920030266US1

5   first instructions for associating an indicator with
6 an object in a heap, wherein the indicator is set to a
7 default state;
8   second instructions, responsive to a first request,
9 for setting indicators for all live objects in the heap
10 from the default state to a live state; and
11   third instructions, responsive to a second request,
12 for collecting information for all objects having
13 indicators in the default state, wherein objects having
14 indicators in the default state are objects with memory
15 leaks.

1 25.   The computer program product of claim 24, wherein
2 the indicator is associated with the object when the
3 object is created.

1 26.   The computer program product of claim 24, wherein
2 the first request and the second request are received
3 from a Java application.

1 27.   A data processing system comprising:
2   a bus system;
3   a memory connected to the bus system, wherein the
4 memory includes a set of instructions; and
5   a processing unit connected to the bus system,
6 wherein the processing unit executes a set of
7 instructions to associate a plurality of indicators with
8 a plurality of objects, wherein the plurality of
9 indicators are set to a first state; set an indicator for

Docket No. RSW920030266US1

10 each live object in the plurality of objects to a second

11 state; and collect data from all objects in the plurality

12 of objects having indicators set to the first state, in

13 response to a request for the data.

1 28. A data processing system comprising:

2   a bus system;

3   a memory connected to the bus system, wherein the

4 memory includes a set of instructions; and

5   a processing unit connected to the bus system,

6 wherein the processing unit executes a set of

7 instructions to associate an indicator with an object a

8 heap, wherein the indicator is set to a default state;

9 set indicators for all live objects in the heap from the

10 default state to a live state, in response to a first

11 request; and collect information for all objects having

12 indicators in the default state, wherein objects having

13 indicators in the default state are objects with memory

14 leaks, in response to a second request.